A short note on the nested-sweep polarized traces method for the 2D Helmholtz equation

Leonardo Zepeda-Núñez[(*)]*, and Laurent Demanet*
Department of Mathematics and Earth Resources Laboratory, Massachusetts Institute of Technology

SUMMARY

We present a variant of the solver in Zepeda-Núñez and Demanet (2014), for the 2D high-frequency Helmholtz equation in heterogeneous acoustic media. By changing the domain decomposition from a layered to a grid-like partition, this variant yields improved asymptotic online and offline runtimes and a lower memory footprint. The solver has online parallel complexity that scales *sublinearly* as $\mathcal{O}\left(\frac{N}{P}\right)$, where N is the number of volume unknowns, and P is the number of processors, provided that $P = \mathcal{O}(N^{1/5})$. The variant in Zepeda-Núñez and Demanet (2014) only afforded $P = \mathcal{O}(N^{1/8})$. Algorithmic scalability is a prime requirement for wave simulation in regimes of interest for geophysical imaging.

INTRODUCTION

Define the Helmholtz equation, in a bounded domain $\Omega \subset \mathbb{R}^2$, with frequency ω and squared slowness $m(\mathbf{x}) = 1/c^2(\mathbf{x})$, by

$$\left(-\triangle - m(\mathbf{x})\omega^2\right) u(\mathbf{x}) = f(\mathbf{x}) \qquad (1)$$

with absorbing boundary conditions. In this note, Eq 1 is discretized with a 5-point stencil, and the absorbing boundary conditions are implemented via a perfectly matched layer (PML) as described by Bérenger (1994). This leads to a linear system of the form $\mathbf{H}\mathbf{u} = \mathbf{f}$. Let N be the total number of unknowns of the linear system and $n = N^{1/2}$ the number of points per dimension. In addition, suppose that the number of point within the PML grows as $\log(N)$. There is an important distinction between:

- the offline stage, which consists of any precomputation involving $\mathbf{H}$, but not $\mathbf{f}$; and
- the online stage, which involves solving $\mathbf{H}\mathbf{u} = \mathbf{f}$ for many different right-hand sides $\mathbf{f}$.

By online complexity, we mean the runtime for solving the system once in the online stage. The distinction is important in situations like geophysical wave propagation, where offline precomputations are often amortized over the large number of system solves with the same matrix $\mathbf{H}$.

Zepeda-Núñez and Demanet (2014) proposed a hybrid between a direct and an iterative method, in which the domain is decomposed in L layers. Using the Green's representation formula to couple the subdomains, the problem is reduced to a boundary integral system at the layer interfaces, solved iteratively, and which can be preconditioned efficiently using a polarizing condition in the form of incomplete Green's integrals. Finally, the integral system and the preconditioner are compressed, yielding a fast application.

Let P for the number of nodes in a distributed memory environment. In Zepeda-Núñez and Demanet (2014), each layer is associated with one node ($P = L$), and the method's online runtime is $\mathcal{O}(N/P)$ as long as $P = \mathcal{O}(N^{1/8})$. The method presented in this note instead uses a nested domain decomposition, with a first decomposition into $L \sim \sqrt{P}$ layers, and a further decomposition of each layer into $L_c \sim \sqrt{P}$ cells, as shown in Fig. 1. The resulting online runtime is now $\mathcal{O}(N/P)$ as long as $P = \mathcal{O}(N^{1/5})$.

The main drawback of the method of polarized traces in Zepeda-Núñez and Demanet (2014) is its offline precomputation that involves computing and storing interface-to-interface local Green's functions. In 3D this approach would become impractical given the sheer size of the resulting matrices. To alleviate this issue, the nested-sweep polarized traces method presented in this paper involves an equivalent *matrix-free* approach that relies on local solves with sources at the interfaces between layers. Given the iterative nature of the preconditioner, solving the local problems naively would incur a deterioration of the online complexity. This deterioration can be circumvented if we solve the local problems inside the layer via the same boundary integral strategy as earlier, in a *nested* fashion. This procedure can be written as a factorization of Green's integrals in block-sparse factors.

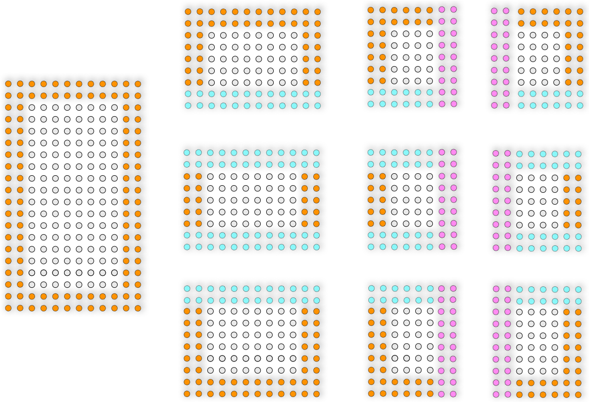


Figure 1: Nested Decomposition in cells. The orange grid-points represent the PML for the original problem, the light-blue represent the artificial PML between layers, and the pink grid-points represent the artificial PML between cells in the same layer.

Finally, the offline complexity is much reduced; instead of computing large Green's functions for each layer, we compute much smaller interface-to-interface operators between the interfaces of adjacent cells within each layer, resulting in a lower memory requirement.

Literature Review

Classical iterative methods require a large number of iterations to converge and standard algebraic preconditioners often fail to improve the convergence rate to tolerable levels (see Erlangga (2008)). Classical domain decomposition methods also fail because of internal reverberations (see Ernst and Gander (2012)). Only recently, the development of a new type of precondtioners have shown that domain decomposition with accurate transmission boundary conditions is the right mix of ideas for simulating propagating high-frequency waves in a heterogeneous medium. To a great extent, the approach can be traced back to the AILU method of Gander and Nataf (2000). The first linear complexity claim was perhaps made in the work of Engquist and Ying (2011), followed by Stolk (2013). Other authors have since then proposed related methods with similar properties, including Chen and Xiang (2013) and Vion and Geuzaine (2014); however, they are often difficult to parallelize, and they usually rely on distributed linear algebra such as Poulson et al. (2013) and Tsuji et al. (2014), or highly tuned multigrid methods such as Stolk et al. (2013) and Calandra et al. (2013). As we put the final touches to this note, Liu and Ying (2015) proposed a recursive sweeping algorithm closely related to the one presented in this note.

In a different direction, much progress has been made on making direct methods efficient for the Helmholtz equation. Such is the case the HSS solver in de Hoop et al. (2011) and hierarchical solver in Gillman et al. (2014). The main issue with direct methods is the lack of scalability to very large-scale problems due to the memory requirements.

To our knowledge, Zepeda-Núñez and Demanet (2014) and this note are the only two references to date that make claims of sublinear $O(N/P)$ online runtime for the 2D Helmholtz equation. We are unaware that this claim has been made in 3D yet, but we are confident that the same ideas have an important role to play there as well.

METHODS

Polarized Traces

This section reviews the method in Zepeda-Núñez and Demanet (2014). Provided that the domain is decomposed in L layers, it is possible to reduce the discrete Helmholtz equation to a discrete integral system at the interfaces between layers, using the Green's representation formula. This leads to an equivalent discrete integral system of the form

$$\underline{\mathbf{M}}\,\underline{\mathbf{u}} = \underline{\mathbf{f}}. \tag{2}$$

The online stage has three parts: first, at each layer, a local solve is performed to build $\underline{\mathbf{f}}$ in Eq. 2; then Eq. 2 is solved obtaining the traces $\underline{\mathbf{u}}$ of the solution at the interfaces betwen layers; finally, using the traces, a local solve is performed at each layer to obtain the solution $\mathbf{u}$ in the volume. The only non-parallel step is solving Eq. 2. Fortunately, there is a physically meaningful way of efficiently preconditioning this system.

The key is to write yet another equivalent problem where local Green's functions are used to perform polarization into one-way components. A wave is said to be polarized as up-going at an interface Γ when

$$0 = \int_{\Gamma} \partial_{\nu_{\mathbf{x}'}} G(\mathbf{x},\mathbf{x}') u^{\uparrow}(\mathbf{x}')dx' - \int_{\Gamma} G(\mathbf{x},\mathbf{x}') \partial_{\nu_{\mathbf{x}'}} u^{\uparrow}(\mathbf{x}')dx',$$

as long as x is below Γ, and vice-versa for down-going waves. These polarization conditions create cancellations within the discrete integral system, resulting in an easily preconditionable system for the polarized interface traces $\underline{\mathbf{u}}^{\uparrow}$ and $\underline{\mathbf{u}}^{\downarrow}$ such that $\underline{\mathbf{u}} = \underline{\mathbf{u}}^{\uparrow} + \underline{\mathbf{u}}^{\downarrow}$. The resulting polarized system is

$$\underline{\underline{\mathbf{M}}}\,\underline{\underline{\mathbf{u}}} = \underline{\underline{\mathbf{f}}}, \qquad \underline{\underline{\mathbf{u}}} = \left(\begin{array}{c} \underline{\mathbf{u}}^{\downarrow} \\ \underline{\mathbf{u}}^{\downarrow} \end{array} \right). \tag{3}$$

The matrix $\underline{\underline{\mathbf{M}}}$ has the structure depicted in Fig. 2. In particular, we write

$$\underline{\underline{\mathbf{M}}} = \left[\begin{array}{cc} \underline{\mathbf{D}}^{\downarrow} & \underline{\mathbf{U}} \\ \underline{\mathbf{L}} & \underline{\mathbf{D}}^{\uparrow} \end{array} \right], \tag{4}$$

where $\underline{\mathbf{D}}^{\downarrow}$ and $\underline{\mathbf{D}}^{\uparrow}$ are, respectively, block lower and upper triangular matrices with identity diagonal blocks, thus trivial to invert via block backsubstitution. In this note, we use the Gauss-Seidel Preconditioner,

$$P^{\mathrm{GS}} \left(\begin{array}{c} \mathbf{v}^{\downarrow} \\ \mathbf{v}^{\uparrow} \end{array} \right) = \left(\begin{array}{c} (\underline{\mathbf{D}}^{\downarrow})^{-1}\mathbf{v}^{\downarrow} \\ (\underline{\mathbf{D}}^{\uparrow})^{-1}\left(\mathbf{v}^{\uparrow} - \underline{\mathbf{L}}(\underline{\mathbf{D}}^{\downarrow})^{-1}\mathbf{v}^{\downarrow}\right) \end{array} \right). \tag{5}$$

The system in Eq. 3 is solved using GMRES preconditioned with P^{GS}.

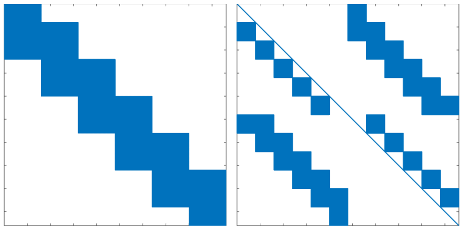


Figure 2: Sparsity pattern of $\underline{\mathbf{M}}$ (left) and $\underline{\underline{\mathbf{M}}}$ (right).

Moreover, one can use an adaptive partitioned low-rank (PLR, also known as $\mathscr{H}$-matrix) fast algorithm for the application of integral kernels, in expressions such as the one above. The implementation details are in Zepeda-Núñez and Demanet (2014).

Nested Solvers

This section now describes the new solver.

One key observation is that the polarized matrix $\underline{\underline{\mathbf{M}}}$ can be applied to a vector in a matrix-free fashion, addressing the offline bottleneck and memory footprint. Each block of $\underline{\underline{\mathbf{M}}}$ is a Green's integral, and its application to a vector is equivalent to sampling a wavefield generated by sources at the boundaries. The application of each Green's integral to a vector $\mathbf{v}$, in matrix-free form, consists of three steps: from $\mathbf{v}$ we form the sources at the interfaces (red interfaces in Fig. 3 left), we perform a local direct solve inside the layer, and we sample the solution at the interfaces (green interfaces in Fig. 3 right).

From the analysis of the rank of off-diagonal blocks of the Green's functions, we know that the Green's integrals can be compressed and applied in $\mathscr{O}(n^{3/2})$ time, but this requires precomputation and storage of the Green's functions. The matrix-free approach does not need expensive precomputations, but would naively perform a direct local solve in the volume, resulting in an application of each Green's integral in $\mathscr{O}(n^2/L)$ complexity (assuming that a good direct method is used at each layer). This note's contribution is to propose "nested" alternatives for the inner solves that result in a lower $\mathscr{O}(L_c(n/L)^{3/2})$ complexity (up to logs) for the application of each Green's integral.

Analogously to Eq. 2, denote the boundary integral reduction of the local problem within each layer as

$$\underline{\mathbf{M}}^\ell \underline{\mathbf{u}}^\ell = \underline{\mathbf{f}}^\ell, \qquad \text{for } \ell = 1, .., L_c; \tag{6}$$

where the unknowns $\underline{\mathbf{u}}^\ell$ are on the interfaces between the L_c cells.

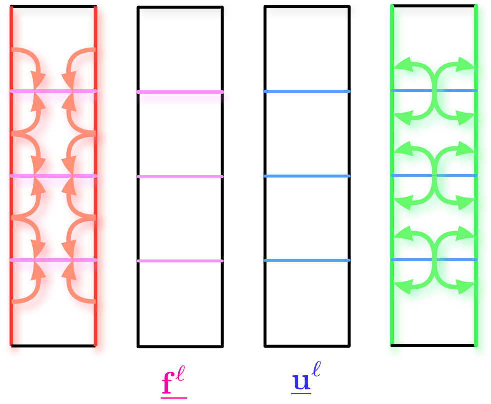


Figure 3: Sketch of the application of the Green's functions. The sources are in red (left) and the sampled field in green (right). The application uses the inner boundaries as proxies to perform the solve.

The nested solver uses the inner boundaries as proxies to perform the local solve inside the layer efficiently. The application of the Green's integral can be decomposed in three steps. Using precomputed Green's functions at each cell we evaluate the wavefield generated from the sources to form $\underline{\mathbf{f}}^\ell$ (from red to pink in Fig. 3 left). This operation can be represented by a sparse block matrix $\mathbf{M}^\ell_f$. Then we solve Eq. 6 to obtain $\underline{\mathbf{u}}^\ell$ (from pink to blue in Fig. 3 right). Finally we use the Green's representation formula to sample the wavefield at the interfaces (from blue to green in Fig. 3), this operation is represented by another sparse-block matrix $\mathbf{M}^\ell_u$.

The algorithm described above leads to the factorization

$$G^\ell = \mathbf{M}^\ell_f \left(\underline{\mathbf{M}}^\ell\right)^{-1} \mathbf{M}^\ell_u, \tag{7}$$

in which the blocks of $\mathbf{M}^\ell_f$ and $\mathbf{M}^\ell_u$ are dense, but highly compressible in PLR form.

Nested polarized traces

To efficiently apply the Green's integrals using Eq. 7, we need to solve Eq. 6 efficiently. A first possibility would be to use the method of polarized traces in a recursive fashion. The resulting recursive polarized traces method has empirically the same scalings as those found in Zepeda-Núñez and Demanet (2014) when the blocks are compressed in partitioned low rank (PLR) form. However, the constants are much larger.

Step	N_{nodes}	Complexity per node
LU factorizations	$O(P)$	$O\left((N/P+\log(N))^{3/2}\right)$
Green's functions	$O(P)$	$O\left((N/P+\log(N))^{3/2}\right)$
Local solves	$O(P)$	$O\left(N/P+\log(N)^2\right)$
Sweeps	1	$O(P(N/P+\log(N)^2)^\alpha)$
Recombination	$O(P)$	$O\left(N/P+\log(N)^2\right)$

Table 1: Complexity of the different steps of the preconditioner, in which α depends on the compression of the local matrices, thus on the scaling of the frequency with respect to the number of unknowns. Typically $\alpha = 3/4$.

A compressed-block LU solver

A better alternative is to use a compressed-block LU solver to solve Eq. 6. Given the banded structure of $\underline{\mathbf{M}}^\ell$ (see Fig. 2), we perform a block LU decomposition without pivoting (without fill-in), which leads to the factorization

$$G^\ell = \mathbf{M}^\ell_f \left(\underline{\mathbf{U}}^\ell\right)^{-1} \left(\underline{\mathbf{L}}^\ell\right)^{-1} \mathbf{M}^\ell_u. \tag{8}$$

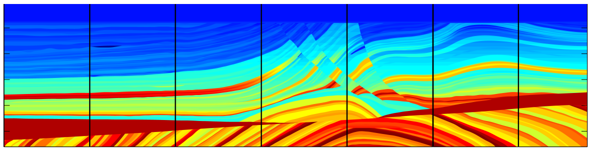
Furthermore, the diagonal blocks of the LU factors are inverted explicitly so that the forward and backward substitutions are reduced to a series of matrix-vector products. Finally, the blocks are compressed in PLR form, yielding a fast solve. This compressed direct strategy is not practical at the outer level because of its prohibitive offline cost, but is very advantageous at the inner level.

The online complexity is comparable with the complexity of the nested polarized traces, but with much smaller constants. The LU factorization has a $\mathscr{O}\left(N^{3/2}/P + \log(N)^{3/2}\right)$ offline cost; which is comparable to the polarized traces method, but with much smaller constants.

COMPLEXITY

Table 1 summarizes the complexities and number of processors at each stage for both methods. For simplicity we do not count the logarithmic factors from the nested dissection; however, we consider the logarithmic factors coming from the extra degrees of freedom in the PML.

If the frequency scales as $\omega \sim \sqrt{n}$, the regime in which second order finite-differences are expected to be accurate, we observe $\alpha = 5/8$; however, we assume the more conservative value $\alpha = 3/4$. The latter is in better agreement with a theoretical analysis of the rank of the off-diagonal blocks of the Green's functions. In such scenario we have that the blocks of $\mathbf{M}^\ell_u$ and $\mathbf{M}^\ell_f$ can be compressed in PLR form, resulting in a fast application in $\mathscr{O}(L_c(n/L+\log(n))^{3/2})$ time, easily parallelizable among L_c nodes. Solving Eq. 6 can be solved using either the direct compressed or the nested polarized traces in $\mathcal{O}(L_c(n/L+\log(n))^{3/2})$ time. This yields the previously advertised runtime of $\mathcal{O}(L_c(n/L+\log(n))^{3/2})$ for each application of the Green's integral.



N	$f[Hz]$	10×2	40×8	100×20
88×425	7.71	**(3)** 0.89	**(3)** 15.6	**(4)** 97.9
175×850	11.1	**(3)** 1.48	**(3)** 17.7	**(3)** 105
350×1700	15.9	**(3)** 2.90	**(3)** 22.1	**(4)** 106
700×3400	22.3	**(3)** 5.58	**(3)** 31.3	**(4)** 126
1400×6800	31.7	**(3)** 10.5	**(3)** 47.9	**(4)** 176

Table 2: Number of GMRES iterations (bold) required to reduce the relative residual to 10^{-5}, along with average execution time (in seconds) of one GMRES iteration using the compressed direct method, for different N and $P = L\times L_c$. The frequency is scaled such that $f = \omega/\pi \sim \sqrt{n}$ and the sound speed is given by the Marmousi2 model (see Martin et al. (2006)).

To apply the Gauss-Seidel preconditioner we need $\mathcal{O}(L)$ applications of the Green's integral, resulting in a runtime of $\mathcal{O}(L\cdot L_c(n/L+\log(n))^{3/2})$ to solve Eq. 3. Using the fact that $L\sim\sqrt{P}$ and $L_c\sim\sqrt{P}$ and adding the contribution of the other steps of the online stage; we have that the overall online runtime is given by $\mathcal{O}(P^{1/4}N^{3/4}+P\log(N)^{3/4}+N/P+\log(N)^2)$, which is sublinear and of the form $O(N/P)$ up to logs, provided that $P=\mathcal{O}(N^{1/5})$.

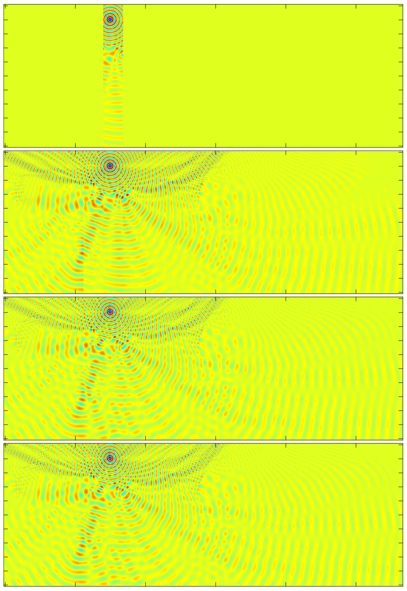


Figure 4: Two iteration of the preconditioner, from top to bottom: initial guess with only local solve; first iteration, second iteration, converged solution.

Finally, the memory footprint is $\mathcal{O}(P^{1/4}N^{3/4}+P\log(N)^{3/4}+N/P+\log(N)^2)$ and the communication cost for the online part is $\mathcal{O}(n\sqrt{P})$, which represents an asymptotic improvement with respect to Zepeda-Núñez and Demanet (2014), in which the storage and communication cost are $\mathcal{O}(PN^{3/4}+N/P+\log(N)^2)$ and $\mathcal{O}(nP)$, respectively.

NUMERICAL RESULTS

Fig. 4 depicts the fast convergence of the method. After a couple of iterations the exact and approximated solution are visually indistinguishable.

Table 2 shows the sublinear scaling for one GMRES iteration, with respect to the degrees of freedom in the volume. We can observe that the number of iterations to converge depends weakly on the frequency and the number of subdomains. Fig. 5 shows the empirical scaling for one global GMRES iteration. We can observe that both methods have the same asymptotic runtime, but with different constants.

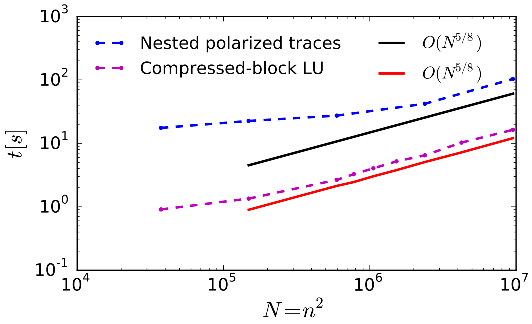


Figure 5: Runtime for one GMRES iteration using the two different nested solves, for $L=9$ and $L_c=3$, and $\omega\sim\sqrt{n}$.

DISCUSSION

We presented an extension to Zepeda-Núñez and Demanet (2014), with improved asymptotic runtimes in a distributed memory environment. The method has sublinear runtime even in the presence of rough media of geophysical interests. Moreover, its performance is completely agnostic to the source.

This algorithm is of especial interest in the context of 4D full wave form inversion, continuum reservoir monitoring, and local inversion. If the update to the model is localized, most of precomputations can be reused. The only extra cost is the refactorization and computation of the Green's functions in the cells with a non null intersection with the support of the update, reducing sharply the computational cost.

We point out that this approach can be further parallelized using distributed algebra libraries. Moreover, the sweeps can be pipelined to maintain a constant load among all the nodes.

ACKNOWLEDGMENTS

The authors thank Total SA for support. LD is also supported by AFOSR, ONR, and NSF.

REFERENCES

Bérenger, J.-P., 1994, A perfectly matched layer for the absorption of electromagnetic waves: Journal of Computational Physics, **114**, 185–200.

Calandra, H., S. Gratton, X. Pinel, and X. Vasseur, 2013, An improved two-grid preconditioner for the solution of three-dimensional Helmholtz problems in heterogeneous media: Numerical Linear Algebra with Applications, **20**, 663–688.

Chen, Z., and X. Xiang, 2013, A source transfer domain decomposition method for Helmholtz equations in unbounded domain: SIAM Journal on Numerical Analysis, **51**, 2331–2356.

de Hoop, M. V., S. Wang, and J. Xia., 2011, On 3D modeling of seismic wave propagation via a structured parallel multifrontal direct Helmholtz solver: Geophysical Prospecting, **59**, 857–873.

Engquist, B., and L. Ying, 2011, Sweeping preconditioner for the Helmholtz equation: Hierarchical matrix representation: Communications on Pure and Applied Mathematics, **64**, 697–735.

Erlangga, Y. A., 2008, Advances in iterative methods and preconditioners for the Helmholtz equation: Archives of Computational Methods in Engineering, **15**, 37–66.

Ernst, O. G., and M. J. Gander, 2012, Why it is difficult to solve Helmholtz problems with classical iterative methods, *in* Numerical Analysis of Multiscale Problems: Springer Berlin Heidelberg, volume **83** *of* Lecture Notes in Computational Science and Engineering, 325–363.

Gander, M., and F. Nataf, 2000, AILU for Helmholtz problems: a new preconditioner based on an analytic factorization: Comptes Rendus de l'Acadmie des Sciences-Series I-Mathematics, **331**, 261–266.

Gillman, A., A. Barnett, and P. Martinsson, 2014, A spectrally accurate direct solution technique for frequency-domain scattering problems with variable media: BIT Numerical Mathematics, 1–30.

Liu, F., and L. Ying, 2015, Recursive sweeping preconditioner for the 3D Helmholtz equation: ArXiv e-prints.

Martin, G., R. Wiley, and K. Marfurt, 2006, An elastic upgrade for Marmousi: The Leading Edge, Society for Exploration Geophysics, **25**.

Poulson, J., B. Engquist, S. Li, and L. Ying, 2013, A parallel sweeping preconditioner for heterogeneous 3D Helmholtz equations: SIAM Journal on Scientific Computing, **35**, C194–C212.

Stolk, C., 2013, A rapidly converging domain decomposition method for the Helmholtz equation: Journal of Computational Physics, **241**, 240–252.

Stolk, C. C., M. Ahmed, and S. K. Bhowmik, 2013, A multigrid method for the Helmholtz equation with optimized coarse grid corrections: ArXiv e-prints.

Tsuji, P., J. Poulson, B. Engquist, and L. Ying, 2014, Sweeping preconditioners for elastic wave propagation with spectral element methods: ESAIM: Mathematical Modelling and Numerical Analysis, **48**, no. 02, 433–447.

Vion, A., and C. Geuzaine, 2014, Double sweep preconditioner for optimized Schwarz methods applied to the Helmholtz problem: Journal of Computational Physics, **266**, 171–190.

Zepeda-Núñez, L., and L. Demanet, 2014, The method of polarized traces for the 2D Helmholtz equation: ArXiv e-prints.